IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s):	George R. Cameron, Hueichian Huang		
Assignee:	3PAR, Inc.		
Title:	Time and Space Efficient Technique for Creating Virtual Volume Copies		
Serial No.:	10/655,951	Filing Date:	September 4, 2003
Examiner:	Shahid Al Alam	Group Art Unit:	2162
Docket No.:	M-8497 US (3PD-M-8497 US)	Confirmation No.	8894

San Jose, California
April 2, 2008

Mail Stop Appeal Brief - Patents
Under Secretary of Commerce for Intellectual Property and Director of the
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, Virginia 22313-1450

APPEAL BRIEF UNDER 37 CFR §41.37(c)

Dear Sir:

Applicant hereby appeals the Final Rejection of claims 1 to 14 in the September 28, 2007 Final Office Action to the Board of Patent Appeals and Interferences. A one-month extension of time is hereby requested, extending the period for response to April 28, 2008.

Please charge the fee of $255.00 for this Appeal Brief as set forth in 37 C.F.R. §41.20(b)(2) to the authorized credit card. Please also charge any amounts underpaid or credit any amounts overpaid to Deposit Account No. 502226.

REAL PARTY IN INTEREST

The real party in interest is the assignee 3PAR, Inc.

RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences.

STATUS OF CLAIMS

All claims 1 to 14 are pending, rejected, and appealed.

STATUS OF AMENDMENTS

The Examiner issued the Final Office Action on September 28, 2007. Applicant did not file any amendments subsequent to the final rejection.

SUMMARY OF CLAIMED SUBJECT MATTER

Embodiments of the invention relate in general to a snapshot tree structure having a base volume, a snapshot descending from the base volume, and another snapshot that descends from the snapshot earlier in the branch.

Claims 1 to 5

Claim 1 recites a computer readable medium for a data storage device encoded with a snapshot tree structure. The snapshot tree structure includes a first branch with the base volume storing a current user data, a first read-only snapshot descending from the base volume, and a second read-only snapshot descending from the first snapshot. The snapshot tree structure is shown in Fig. 8 and described in paragraphs [0090] and [0091] as follows:

> [0090] Each snapshot volume maintains data and tables for an associated snapshot of the base virtual volume. As such, for snapshot tree 2000, the snapshot volumes may be considered to "descend" from a base virtual volume 2200. Any of the snapshot volumes can be accessed to obtain data that was written at a prior time. A snapshot volume can be either a read-only (R/O) snapshot volume (or ROSS) or a read/write (R/W) snapshot volume (or RWSS). A ROSS presents a constant view of the data in a virtual volume at a specific time. After creation of a ROSS, data can be read from but not written into the ROSS. A RWSS descends from a ROSS (e.g., a parent snapshot volume) and may serve to hold modifications to the parent ROSS. A RWSS can be read and

written like a base virtual volume. As such, a RWSS can be viewed as a writable/modifiable version of its parent ROSS. As shown, snapshot volumes 2106, 2204, 2210, 2212, 2206, 2308, and 2310 are ROSSes, and snapshot volumes 2104, 2304, and 2306 are RWSSes. Each of the RWSSes may have one or more descending ROSSes. As can be seen, for example, a RWSS 2306 can descend from a ROSS 2308 of another RWSS 2304.

[0091] The snapshot volumes may be grouped in branches. A branch is made up of a read-write volume (either base virtual volume or RWSS) as its base and one or more read-only snapshot volumes maintained in a time-ordered link attached to the read-write volume. Thus, referring to FIGURE 8 for example, a branch can be the base virtual volume 2200 and a sequence of read-only snapshots volumes, such as the ROSSes 2204, 2210, 2212, and 2206. A branch may also be a read/write snapshot volume, such as a RWSS 2304 and one or more read-only snapshot volumes, such as the ROSSes 2308 and 2310. A new branch can be created by adding a read-write snapshot volume to a read-only snapshot volume, after which read-only snapshot volumes can be added to grow the branch. For any given branch, the snapshot volumes extend from oldest to most recent. For example, in the branch comprising base volume 2200 and snapshot volumes 2204, 2210, 2212, and 2206, snapshot volume 2206 is the oldest (created earliest in time) while snapshot 2204 is the most recent (created last in time).

Specification, ¶¶ [0090] and [0091] (emphasis added).

The first read-only snapshot is created at a first time and stores a first data of the base volume at the first time before the first data is modified in the base volume. Specification, ¶¶ [0061], [0090], and [0092]. The second read-only snapshot is created at a second time earlier than the first time and stores a second data of the base volume at the second time before the second data is modified in the base volume. Id.

Claim 1 further recites code for managing the snapshot tree structure to provide point-in-time backups of a base volume. The code includes instructions to retrieve data from the snapshot tree structure and to transmit the retrieved data to a host device. Specification, ¶¶ [0030], [0059], and [0089].

Claims 2 to 5 depend from claim 1 and are patentable for at least the same reasons as claim 1.

Claims 6 to 10

Claim 6 is a method claim that parallels apparatus claim 1. Claim 6 recites a method for a data storage device to store snapshots that provide point-in-time backups of a base volume using a

snapshot tree structure. The method includes creating a first branch of the snapshot tree structure by creating the base volume, creating a first read-only snapshot descending from the base volume at a first time, creating a second read-only snapshot descending from the base volume at a second time later than the first time, and inserting the second read-only volume between the base volume and the first read-only snapshot so the first read-only snapshot now descends from the second read-only snapshot. Specification, ¶ [0091]. The base volume stores current user data, the first read-only snapshot stores a first data in the base volume at the first time before the first data is modified in the base volume, and the second read-only snapshot stores a second data in the base volume at the second time before the second data is modified in the base volume. Specification, ¶¶ [0061], [0090], and [0092]. The method further includes retrieving data from the snapshot tree structure and transmitting the retrieved data to a host device. Specification, ¶¶ [0030], [0059], and [0089].

Claims 7 to 10 depend from claim 6 and are patentable for at least the same reasons as claim 6.

Claims 11 to 14

Claim 11 recites a method to retrieve a point-in-time backup of a base volume by reading a data block from a snapshot tree structure. Similar to claim 1, the snapshot tree structure includes the base volume, a first read-only snapshot descending from the base volume, and a second read-only snapshot descending from the first snapshot. Specification, ¶ [0091]. The method includes searching for the data block in the second snapshot and, when the data block is not found in the second snapshot, following a link in the second snapshot to the first snapshot and searching for the data block in the first snapshot. Specification, ¶¶¶ [0094] and [0095]. The method further includes transmitting the data block to a host device after the data block is found. Specification, ¶¶ [0030], [0059], and [0089].

Claims 12 to 14 depend from claim 11 and are patentable for at least the same reasons as claim 11.

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

The Examiner rejected claims 1 to 14 under 35 U.S.C. § 102(b) as being anticipated by the reference "File System Design for an NFS File Server Appliance" by Dave Hitz et al. (hereafter "Hitz et al.").

ARGUMENTS

Claims 1 to 5

Claim 1 recites a snapshot tree structure having a first branch with a base volume, a first read-only snapshot descending from the base volume, and a second read-only snapshot descending from the first snapshot. The base volume stores a current user data. The first read-only snapshot is created at a first time and stores a first data of the base volume at the first time before the first data is modified in the base volume. The second read-only snapshot is created at a second time earlier than the first time and stores a second data of the base volume at the second time before the second data is modified in the base volume.

While Hitz et al. discloses that the WAFL file system keeps up to 20 snapshots online to provide access to old versions of files, it does not disclose that any snapshot descends from another snapshot. The Examiner cited Figs. 2, 3, and 4 of Hitz et al. to show a snapshot descending from another snapshot. Applicant respectfully traverses.

Fig. 2 shows the structure of the WAFL file system's tree of blocks. The tree of blocks consists of a root inode containing inodes that describe the rest of the files in the file system. The files are made up of individual blocks where larger files have additional layers of indirection between the inode and the actual data blocks. As one can clearly see, Fig. 2 does not show any snapshot that descends from another snapshot.

Fig. 3 shows that the WAFL file system creates a snapshot of a tree of blocks by duplicating the root inode. Specifically, the duplicated root inode becomes the root of a tree of blocks representing the snapshot. When a user modifies a data block, the active file system points to a new data block while the snapshot references the original data block. As one can clearly see, Fig. 3 does not show any snapshot that descends from another snapshot.

Fig. 4 shows the creation of the snapshot in Fig. 3 with more detail. Specifically, when a data block is modified, its contents are written to a new location and the pointers in the block's ancestors are updated and written to new locations as well. As one can clearly see, Fig. 4 does not show any snapshot that descends from another snapshot. For the above reasons, claim 1 is patentable over Hitz et al. as it does not disclose any snapshot that descends from another snapshot.

Claims 2 to 5 depend from claim 1 and are patentable over Hitz et al. for at least the same reasons as claim 1.

Claims 6 to 10

Claim 6 is a method claim that parallels apparatus claim 1. Thus, amended claim 6 is patentable over Hitz et al. for at least the same reasons as claim 1.

Claims 7 to 10 depend from claim 6 and are patentable over Hitz et al. for at least the same reasons as claim 6.

Claims 11 to 14

Claim 11 recites searching for a data block in a second snapshot, which descends from a first snapshot, which descends from a base volume, and, when the data block is not in the second snapshot, searching for the data block in the first snapshot. As described above, Hitz et al. does not disclose a snapshot descending from another snapshot. Accordingly, claim 11 is patentable over Hitz et al.

Claims 12 to 14 depend from claim 11 and are patentable over Hitz et al. for at least the same reasons as claim 11.

CONCLUSION

Applicant respectfully submits the Examiner has failed to show that the cited references disclose all the recited elements of claims 1 to 14. Accordingly, Applicant requests the rejections of claims 1 to 14 to be reversed.

Respectfully submitted,

/David C Hsia/

David C. Hsia
Attorney for Applicant(s)
Reg. No. 46,235

Patent Law Group LLP
2635 North First St., Ste. 223
San Jose, California 95134
408-382-0480x206

CLAIMS APPENDIX

Claim 1: A computer readable medium for a data storage device encoded with a snapshot tree structure and code for managing the snapshot tree structure to provide point-in-time backups of a base volume, wherein:

the snapshot tree structure comprises:

a first branch, comprising:

the base volume storing a current user data;

a first read-only snapshot descending from the base volume, the first read-only snapshot being created at a first time, the first read-only snapshot storing a first data of the base volume at the first time before the first data is modified in the base volume; and

a second read-only snapshot descending from the first snapshot, the second read-only snapshot being created at a second time earlier than the first time, the second read-only snapshot storing a second data of the base volume at the second time before the second data is modified in the base volume; and

the code comprises instructions to retrieve data from the snapshot tree structure and transmitting the retrieved data to a host device.

Claim 2: The snapshot tree structure of claim 1, further comprising:

a second branch, comprising a first read-write snapshot descending from one of the first and the second read-only snapshots.

Claim 3: The snapshot tree structure of claim 2, wherein the second branch further comprises a third read-only snapshot descending from the first read-write snapshot, the third read-only snapshot being created at a third time, the third read-only snapshot storing a third data of the first read-write snapshot at the third time before the third data is modified in the first read-write snapshot.

Claim 4: The snapshot tree structure of claim 3, further comprising:

a third branch, comprising a second read-write snapshot descending from the third read-only snapshot.

Claim 5: The snapshot tree structure of claim 4, wherein the third branch further comprises a fourth read-only snapshot descending from the second read-write snapshot, the fourth read-only snapshot being created at a fourth time, the fourth read-only snapshot storing a fourth data of the second read-write snapshot at the fourth time before the fourth data is modified in the read read-write snapshot.

Claim 6: A method for a data storage device to store snapshots that provide point-in-time backups of a base volume using a snapshot tree structure, the method comprising:

creating a first branch, comprising:

creating the base volume storing a current user data;

creating a first read-only snapshot descending from the base volume, the first read-only snapshot being created at a first time;

storing in the first read-only snapshot a first data of the base volume at the first time before the first data is modified in the base volume;

creating a second read-only snapshot descending from the base volume, the second read-only snapshot being created at a second time later than the first time;

storing in the second read-only snapshot a second data of the base volume at the second time before the second data is modified in the base volume; and

inserting the second read-only snapshot between the base volume and the first read-only snapshot, wherein the first read-only snapshot now descends from the second read-only snapshot; and

retrieving data from the snapshot tree structure and transmitting the retrieved data to a host device.

Claim 7: The method of claim 6, further comprising:

creating a second branch, comprising creating a first read-write snapshot descending from one of the first and the second read-only snapshots.

Claim 8: The method of claim 7, wherein said creating a second branch further comprises creating a third read-only snapshot descending from the first read-write snapshot, the third read-only snapshot being created at a third time, the third read-only snapshot storing a third data of the first read-write snapshot at the third time before the third data is modified in the first read-write snapshot.

Claim 9: The method of claim 8, further comprising:

creating a third branch, comprising creating a second read-write snapshot descending from the third read-only snapshot.

Claim 10: The method of claim 9, wherein said creating a third branch further comprises creating a fourth read-only snapshot descending from the second read-write snapshot, the fourth read-only snapshot being created at a fourth time, the fourth read-only snapshot storing a fourth data of the second read-write snapshot at the fourth time before the fourth data is modified in the read read-write snapshot.

Claim 11: A method for a data storage device to retrieve a point-in-time backup of a base volume by reading a value of a data block from a snapshot tree structure having the base volume, a first snapshot descending from the base volume, and a second snapshot descending from the first snapshot, the method comprising:

searching for the data block in the second snapshot;

if the data block is not found in the second snapshot:

following a link in the second snapshot to the first snapshot; and

searching for the data block in the first snapshot; and

transmitting the data block to a host device after the data block is found.

Claim 12: The method of claim 11, wherein the first and the second snapshots are read-only snapshots.

Claim 13: The method of claim 11, wherein the first snapshot is a read-only snapshot and the second snapshot is a read-write snapshot.

Claim 14: The method of claim 11, further comprising:

if the data block is not found in the first snapshot:

following a link in the first snapshot to the base volume; and

reading the data block from the base volume.

EVIDENCE APPENDIX

None

RELATED PROCEEDINGS APPENDIX

None.